REMARKS

The present response is intended to be fully responsive to all points of rejection raised by the Examiner and is believed to place the application in condition for allowance. Favorable reconsideration and allowance of the application is respectfully requested.

Claims 1-64 are pending in this case. Claims 1-64 have been rejected under 35 U.S.C. § 103(a). Independent claims 1, 21, 41, 61 and 63 and dependent claims 2-3, 8-20, 22-23, 28-30, 35-40, 42-43, 48-50, 55-60, 62,64 have been amended.

With respect to the Examiner's 35 U.S.C. § 103(a) rejections, Applicant has reviewed the cited art and respectfully submits that the art fails to disclose or suggest the Applicant's claimed invention. Therefore, Applicant respectfully traverses and requests favorable reconsideration.

Response to 35 U.S.C. § 103(a) Rejections

The Examiner rejected claims 1-64 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,901,225 ("Ireton et al.") in view of U.S. Patent No. 6,247,168 ("Green).

While continuing to traverse the Examiner's rejections, Applicant, in order to expedite the prosecution, has chosen to clarify and emphasize the crucial distinctions between the present invention and the devices of the patents cited by the Examiner. Specifically, claim 1 has been amended to include a method of securely downloading and installing a patch program in a plurality of computing devices, each computing device having a processor, program memory and patch memory, the method comprising the steps of encrypting the entire contents of the patch program using a shared key to generate a first encrypted patch program, the shared key being known to a plurality of computing devices, transmitting the first encrypted patch program to a plurality of the computing devices over a nonsecure communications channel, receiving the first encrypted patch program at a computing device, decrypting the first encrypted patch program utilizing the shared key to generate a first clear patch program, re-encrypting the entire contents of the clear patch program utilizing a unique key known only to and hardwired into that particular computing device and storing a resultant second encrypted patch program in memory, upon subsequent reset, retrieving the second encrypted patch program from the memory and decrypting the second encrypted patch program utilizing the unique key to generate a second clear patch program and loading the second clear patch program into the patch memory for execution by the computing device.

Green teaches a tool for programming non-volatile memory that is embedded in the form of an object in a programmable controller module to be used to transfer a firmware program to a plurality of different modules connected by a common network. The system comprises an encryption/decryption program. To update the firmware of a target module, the encryption program encrypts the serial number of the target module. The serial number is unique to the target module and distinguishes the target module from every other individual module in the entire PLC product line. The "approved" serial number of the target module is encrypted using the encryption key to generate an encrypted serial number which is then downloaded to the processor module. The encrypted serial number is decrypted using the key. The decrypted serial number downloaded with the patch is compared to the serial number of the target module. The update is performed only if there is a match.

Ireton et al. teaches a system and method for performing software patches for embedded system devices in which the firmware resides in non-alterable storage of the device. An encryption mechanism is disclosed for increasing the security of the embedded system comprising the device. The patch is encrypted prior to storage in the external memory and decrypted as the patch is loaded into the device in order to recover the original bit sequence of the patch.

It is submitted that the system of Green uses the unique serial of the module in the encryption of the patch **before** it is downloaded to the module. In this system, the serial number of the target module must be supplied to the firmware producer **beforehand.** The firmware producer then encrypts the serial number with a key known to the client (embedded in the client module) for transmission to the module. Green does not indicate whether the key used in the encryption of the serial number is shared or unique to the requesting module. Further, Green apparently **only** encrypts the serial number of the firmware and **not** the contents of the firmware itself.

In contrast, the scheme of the present invention uses a **shared key** known to a plurality of devices to encrypt the **entire** contents of the patch program. The encrypted patch program is then broadcast to all or a subset of all the devices. No information unique to any particular device is used in preparing the patch program for transmission to the devices as is done by Green. All devices that have knowledge of the shared key can receive and decrypt the patch program. The output of the decryption step is the original clear patch program. After decryption is complete, the integrity of the clear patch program is verified.

If the verification is successful, the clear patch program is re-encrypted using a **unique key** which is unique across the entire systems. Each individual computing device is assigned its own unique key and is burned (i.e. hardwired) into an integrated circuit (i.e. the processor or associated

memory) within the computing device. The clear patch program is re-encrypted using the unique key associated with that particular computing device and the result is a second encrypted patch program. This second encrypted patch program is stored in local non-volatile memory.

Subsequently, after each reset (i.e. reboot, startup, power-up, etc.) the second encrypted patch program is retrieved from memory and decrypted using the unique key known only to that particular computing device. This second decryption step generates a second clear patch program which is written into special patch data within the computing device for execution by the processor. The advantage of this scheme of the present invention is that at no time is the program patch exposed externally with the consequent risk of comprise by hackers. The re-encryption step is important as the number of devices using the same shared key could be large. This is why the patch program is re-encrypted with the device's unique key since only that device is able to decrypt it in the event it is compromised by hacking. Thus it is important to perform the **two stage** encryption/decryption process in order to secure the program patch while being stored by each computing device. None of these features are anticipated, taught nor suggested by the Green reference.

It is submitted that the method of Ireton et al. uses a "one time pad" as a key used in the encryption mechanism. According to Ireton et al., the temporary key may generated using "ionosphere scattering", a "bit sequence of software program instructions" or a string of random numbers and used to encrypt a message and is used on a "one time only" basis to encrypt the software. See col. 10, lines 33-39. Further, the method does not indicate whether the key used is unique to a particular embedded system or common to many.

In contrast, the secure download and storage scheme of the present invention is operative to re-encrypt the patch program, after decryption by the shared key, using a **permanent key** that is **hardwired** (i.e. burned) into the processor IC or other chip means of the computing device. This unique permanent key is known only to that particular computing device and is never shared with any other device. Further, this unique key is repeatedly used for all program patches received by the computing device. This feature is neither taught nor suggested by Ireton et al.

Applicant respectfully submits that the Examiner has failed to show that one of ordinary skill in the art would have been motivated to modify Ireton et al. in view of Green to arrive at the claimed invention because there is no suggestion made by Ireton et al. or Green to use a first shared key to encrypt the patch before transmitting it to a plurality of computing devices and on each individual device to use a second unique key to re-encrypt the patch once received and decrypted using the first shared key.

Applicants submit that the combination of Ireton et al. and Green would not result in the claimed invention. The Examiner has improperly combined Ireton et al. and Green in an attempt to arrive at the claimed invention. The combination suggested by the Examiner fails to teach or suggest all the claims limitations. The combination of Ireton et al. and Green fails to teach using a first shared key to encrypt the patch before transmitting it to a plurality of computing devices and on each individual device to using a second unique key to re-encrypt the patch once received and decrypted using the first shared key.

It is believed that amended independent claims 1, 21, 41 and new independent claims 61, 63 overcome the Examiner's § 103(a) rejection based on the Ireton et al. and Green references. Because Ireton et al. and Green do not anticipate or suggest claims 1, 21, 41, 61, 63 as discussed above, then claims 2-20, 22-40, 42-60, 62, 64 are allowable as well. The Examiner is respectfully requested to withdraw the rejection based on § 103(a).

Conclusion

In view of the above amendments and remarks, it is respectfully submitted that independent claims 1, 21, 41, 61, 63 and hence dependent claims 2-20, 22-40, 42-60, 62, 64 are now in condition for allowance. Prompt notice of allowance is respectfully solicited.

In light of the Amendments and the arguments set forth above, Applicant earnestly believes that they are entitled to a letters patent, and respectively solicit the Examiner to expedite prosecution of this patent applications to issuance. Should the Examiner have any questions, the Examiner is encouraged to telephone the undersigned.

Customer Number: 25937

Respectfully submitted,

ZARETSKY & ASSOCIATES PC

By: _____

Howard Zaretsky
Reg. No. 38,669
Attorney for Applicants

Zaretsky & Associates PC
8753 West Runion Dr
Peoria AZ 85382-6412
Tel.: 623-362-2585